

Методика разработки гипертекстовых средств учебного назначения для поддержки принятия управленческих решений

Порошин А. Н.¹

¹ pan@yandex.ru

¹ Санкт-Петербургский государственный университет, С-Петербург, Россия

Предложена методика разработки гипертекстовых компьютерных средств обучения в области профессионального образования с использованием графического языка Дракон и системы управления сайтом Wikipad. На основе данной методики возможна эффективная реализация обучающих тренажеров для принятия управленческих решений, например, в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд.

Ключевые слова: гипертекст, методика, образовательные ресурсы, управленческие решения

Введение

Сегодняшний уровень развития общества характеризуется возрастающей потребностью в профессиональных управленческих кадрах, обладающих необходимыми навыками, приобретение которых в реальных условиях часто затруднено или даже невозможно. Это является причиной разработки и внедрения специализированных учебных средств на базе современных информационно-коммуникационных технологий, прежде всего, компьютерных тренажеров, систем поддержки принятия решений и других практикоориентированных компьютерных средств обучения (КСО), предназначенных в основном для сферы профессионального и дополнительного образования.

Отличительной особенностью подобных систем является наличие модели объекта управления, причем качество ее реализации, заданное вычислительными алгоритмами, возможности удобного и наглядного отображения результатов расчетов, дружественный интерфейс, доступ к сетевой среде, внесение элементов деловой игры – все перечисленные аспекты имеют первостепенное значение для мотивации обучаемых, снижения их утомления и достижения высокой эффективности при работе. И если ранее такие КСО создавались исключительно в виде десктопных приложений, то в последнее время более прогрессивными считаются программные продукты, реализованные как кроссплатформенные веб-приложения.

Отметим, что на рынке программных средств имеются различные продукты учебного назначения, реализованные как веб-приложения, подчас весьма развитые, например системы управления электронными курсами «Moodle», «Blackboard», «Гиперметод» и другие. Однако они нацелены больше на использование в системах дистанционного обучения, и, благодаря многообразию типов информационных ресурсов и элементов обучающих курсов (текст, веб-страница, лекция, тест, форум, глоссарий и т. д.) являются интегрированными информационно-образовательными средами для теоретической подготовки и тестирования учащихся. И качественное использование указанных систем предполагает немалые расходы на их приобретение, развертывание и сопровождение, наличие серьезных аппаратных серверных ресурсов, хорошую подготовку технического персонала и специальное обучение создателей контента и тьюторов.

В настоящей работе предлагается методика эффективной разработки компактных малобюджетных гипертекстовых КСО, которые могут быть созданы одним педагогом или небольшой группой специалистов-предметников при минимальном участии ИТ-специалиста, а в идеале – без привлечения такового. Основными ее элементами являются ранее апробированные и свободно распространяемые программные продукты и технологии, которые, в силу ряда причин, до настоящего времени малоизвестны в широких кругах специалистов сферы образования. Прежде всего к ним относятся рассматриваемые ниже отечественный графический язык Дракон, программная система Drakon Editor и система управления сайтом Wikipad, которые при комплексном использовании обеспечивают весь цикл разработки.

Этот цикл начинается с построения алгоритмической модели КСО средствами графического языка с ее последующим автоматическим преобразованием в текст на языке программирования гипертекстовых ресурсов на стороне клиента (JavaScript). Далее полученная программа может быть либо непосредственно представлена в тексте веб-страницы, либо внедрена в качестве блока кода в систему управления сайтом Wikipad, которая играет роль оболочки, интегрирующей контент КСО и предоставляющей необходимый интерфейс с пользователем, средства редактирования модели и другие базовые и сервисные функции. В этом последнем случае в зависимости от режима использования Wikipad может выступать и в качестве удобного авторского средства (authoring tool) создания учебного контента для педагога-предметника, и как собственно КСО для пользователя-учащегося.

Описание алгоритмической модели средствами языка Дракон

Рассмотрим более подробно этап формализации модели и средство его поддержки – язык Дракон¹ [1, 2]. Этот отечественный визуальный алгоритмический язык был создан в рамках космической программы «Буран» в период с 1986 по 1996 годы. Основной задачей разработчиков было создание единого универсального языка программирования и моделирования, который был бы способен заменить специализированные языки за счет своей доступности и мощности. Дракон построен путём формализации, эргономизации и структуризации блок-схем алгоритмов, описанных стандартами ГОСТ 19.701–90 и ISO 5807–85 и

¹ Дружелюбный Русский Алгоритмический язык, Который Обеспечивает Наглядность.

по сравнению с ними имеет ряд серьезных преимуществ.

Являясь математически строгим и топологически однозначным, он наглядно показывает главный маршрут алгоритма, позволяя легко различать его побочные ветви, а также поддерживает описание процессов реального времени. Наконец, с точки зрения эргономики, Дракон предписывает осмысленное направление движения глаз человека в поле чертежа (по вертикали и горизонтали), за счет чего многократно улучшается «понимаемость» алгоритма и минимизируется вероятность появления ошибок, в то время как блок-схемы, наоборот, зачастую провоцируют их, особенно при увеличении алгоритмической сложности.

Язык содержит только графические элементы (иконки-пиктограммы, линии связи и т. п.), а текст (управляющие слова, т. е. традиционные операторы, как в паскале, бейсике и т. п. языках) отсутствует. Отсутствие необходимости запоминать специальные ключевые слова и их семантику чрезвычайно удобно именно для специалистов-непрограммистов, поскольку это значительно облегчает усилия по формализации задачи. Можно отметить и то обстоятельство, что кроме Дракона в настоящее время практически не существует других языков, обладающих такими же качествами.

Первоначально описание алгоритмической модели может быть выполнено как традиционным способом, на бумаге, так и сразу же непосредственно в среде системы Drakon Editor, позволяющей вводить и редактировать графическое описание алгоритма (включающего арифметические и логические выражения) на основе стандартных иконок Дракона. Естественным продолжением этого шага является автоматическое преобразование полученной схемы алгоритма в текст на выбранном базовом языке. В настоящее время Drakon Editor в стандартной поставке поддерживает ряд таких языков программирования, как C/C++, C#, D, Erlang, Java, Javascript, Lua, Python, Tcl. Однако при необходимости с помощью дополнительного плагина можно обеспечить кодогенерацию и для других языков, что потребует, естественно, нескольких часов работы программиста. Предлагаемая методика рассчитана на создание КСО в виде веб-приложения, что однозначно предопределяет и выбор языка Javascript в качестве базового.

Система управления сайтом Wikipad как авторское средство создания контента

После того, как получен программный код на базовом языке, реализующий функционал разрабатываемого КСО, он может быть непосредственно вставлен в HTML-код одной или нескольких гипертекстовых страниц, формируя тем самым учебное веб-приложение, готовое для тестирования и дальнейшего использования. Однако при этом функциональные возможности такого приложения обычно невелики и, кроме того, подготовка его текста на языке гипертекстовой разметки представляет для неспециалиста непростую задачу. В этой ситуации гораздо удобнее использовать систему управления сайтом (Content Management System, CMS), которая играет роль оболочки и предоставляет базовый функционал для обработки контента, включая и возможности расширения системы за счет дополнительных модулей-плагинов.

В рамках представленной работы в качестве CMS выбрана система Wikipad [4], лицензия на которую разрешает ее свободное распространение и доработку. Входным языком Wikipad является ограниченный естественный язык, что характерно для wiki-подхода и позволяет создать на его основе встроенный предметно-ориентированный язык (domain specific language, DSL) описания педагогического сценария.

Как указано выше, система функционирует в двух режимах: автора (педагога) и пользователя (учащегося). В первом режиме преподавателю доступны встроенные средства формирования гипертекстовых страниц, с помощью которых и реализуется педагогический сценарий. При этом допустимо использовать все возможности базовой wiki-разметки (создание абзацев, списков, таблиц, гиперссылок, изображений), а также средства встраиваемого языка для программного доступа к гипертекстовым объектам (на основе русскоязычной мнемоники). Типовые элементы модели, а также сервисные возможности, например, организация диалога, могут быть поддержаны макросредствами, которые также встроены в DSL.

Из указанных элементов может быть сформировано содержание интерактивной динамической гипертекстовой страницы, включающей, помимо текста и изображений, активные элементы, например выбор по условию, многократное выполнение последовательности действий, ввод и проверку введенных данных и др. Вид отображаемых страниц зависит от применяемых шаблонов, которые могут разрабатываться в зависимости от особенностей представления информации на конкретной странице. С целью поддержки ряда специализированных функций (например, вывода деловой графики) допустимо подключение сторонних библиотек кода, описание которых также может быть задано в файлах шаблонов.

Режим учащегося является основным и позволяет работать с системой как с обычным гипертекстовым ресурсом в локальном (на отдельном компьютере) или серверном (например, с подключением к компьютерной сети учебного заведения) вариантах. В этом режиме реализуется описанный преподавателем педагогический сценарий, поддерживается диалоговое взаимодействие посредством форм ввода, отображаются результаты расчетов, выдаются подсказки и комментарии системы. Таким образом, в одном продукте объединен как функционал авторского инструментального средства разработки учебного контента, позволяющий описывать достаточно сложные модели, так и удобная кросс-платформенная среда обучения, причем для взаимодействия с системой и автору, и учащемуся достаточно обладать навыками работы с программой-браузером.

Следует подчеркнуть такие положительные характеристики созданной системы как дружелюбность, компактность, возможность запуска как на локальной машине, так на сервере, легкость освоения пользо-

вателем, простота развертывания и поддержки. В основе ее концепции лежит дальнейшее развитие понятия многослойной гипертекстовой среды, соединившей в себе наряду с упоминаемыми в [5] структурно-семантическими слоями, такие виртуальные слои, как слой разработки, демонстрации, сценарный, графический и макроописаний.

Заключение

Таким образом, первые два этапа методики включают четкую постановку задачи и ее формализацию с использованием табличных или аналитических моделей. Затем разрабатывается педагогический сценарий, а на его основе средствами языка DSL производится детализация этого сценария на уровне отдельных учебных блоков, в результате чего получается множество гипертекстовых страниц, соответствующих указанным блокам. Заключительными этапами являются тестирование системы, дополнение шаблонами и макроопределениями, документирование и пробная эксплуатация.

На основе предложенной методики разработки практикоориентированных интерактивных гипертекстовых образовательных ресурсов проводится работа по созданию модельных тренажеров, предназначенных для отработки навыков принятия управленческих решений в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд и ориентированных на обработку преимущественно текстово-графической информации, характерной именно для экономических приложений.

Отметим, что благодаря наличию предметно-ориентированного языка практически все основные этапы разработки тренажера могут быть выполнены преподавателями, владеющими основами алгоритмизации в среде Дракона, т. е. специалистами в конкретной предметной области. Исключение могут составлять реализация сложных процедур вывода и подготовка отдельных шаблонов, а также программирование специальных функций, необходимых для расширения системы.

Список литературы

1. *Паронджанов В. Д.* Как улучшить работу ума: Алгоритмы без программистов – это очень просто! М. : Дело, 2001. 360 с.
2. *Паронджанов В. Д.* Учись писать, читать и понимать алгоритмы. М.: ДМК Пресс, 2014. 520 с.
3. Программа DRAKON Editor. [Электронный ресурс] / С. Б. Митькин. URL: <http://drakon-editor.sourceforge.net/> (дата обращения: 20.05.2014).
4. Wikipad. [Электронный ресурс]: Web-studio «Cherry-Design» / М. Г. Мельников. URL: <http://www.cherry-software.spb.ru/> (дата обращения: 20.05.2014).
5. *Демин И. С.* Многослойные гипертекстовые среды: автореферат дис. ... доктора экон. наук: 08.00.13 / Финансовая акад. при Правительстве РФ. М., 2004.